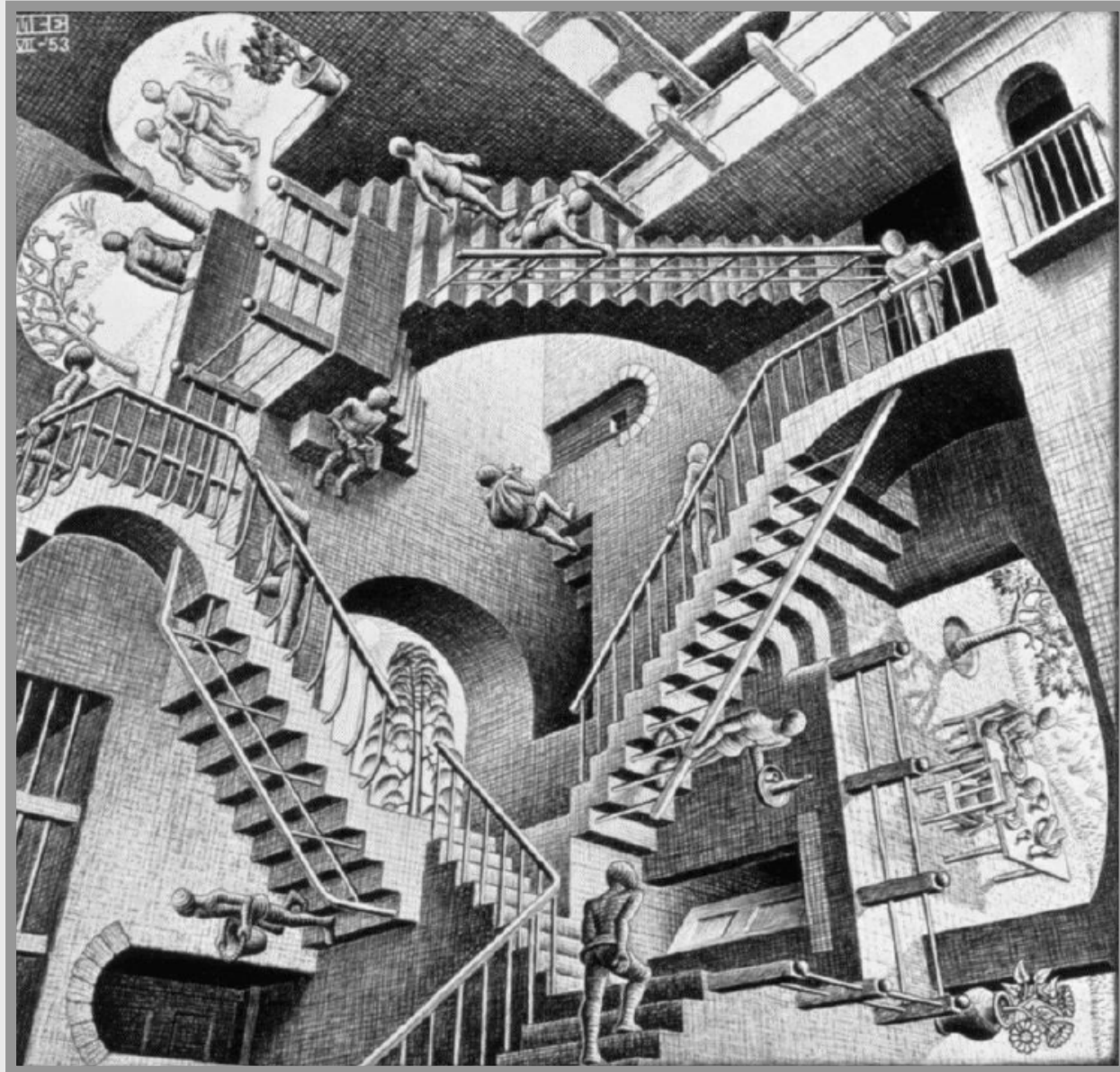


Wissens- basierte Diagnostik

Kap. 7:
Case Based Reasoning Teil 1

Dr. Norbert Waleschkowski
Fachbereich Informatik

Vorlesung Master-Studiengang
Wintersemester 2009/10



Was ist CBR (Case-Based Reasoning) ?

- „Case-based reasoning is (...) reasoning by remembering.“ (Leake 1996)
- „A case-based reasoner solves new problems by adapting solutions that were used to solve old problems.“ (Riesberg & Schank 1998).
- „Case-based reasoning is a new approach to problem solving and learning.“ (Aamodt & Plaza 1994)
- „Case-based reasoning is both (...) the ways people uses cases to solve problems and the ways we make machines use them.“ (Kolodner 1993)

Case-Based Reasoning in a nutshell

- Store previous experience (cases) in memory.
- To solve new problems:
 - ◆ retrieve similar experience about similar situations from the memory
 - ◆ reuse the experience in the context of the new situation: complete or partial reuse, or adapt according to differences
 - ◆ store new experience in memory (learning)

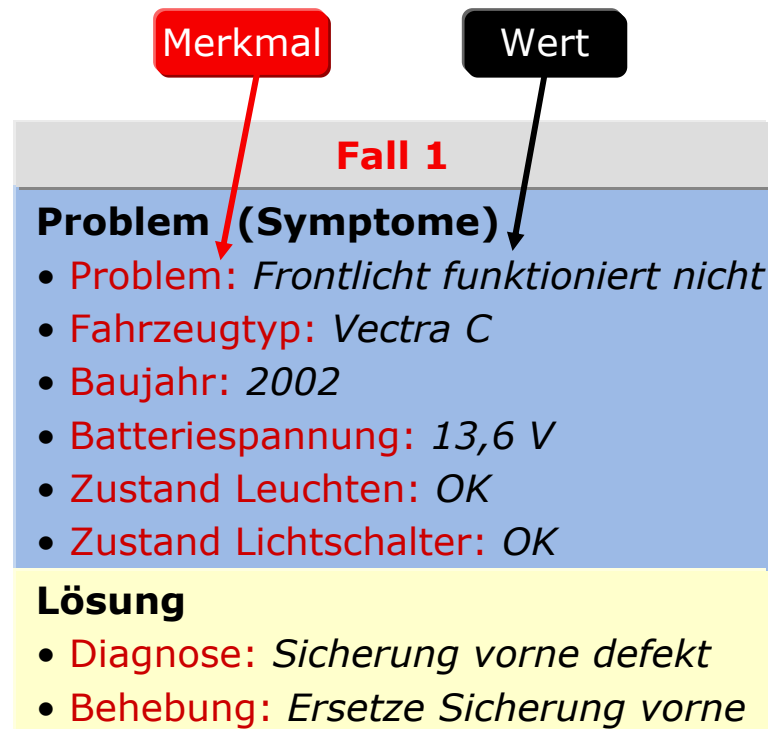
Ein einfaches Beispiel

- Fahrzeugdiagnose:
 - ◆ Wahrnehmung/Beobachtung von Symptomen (z.B. Motor startet nicht)
 - ◆ Ermittlung von Messwerten (z.B. Batteriespannung = 6.3 V).
 - ◆ ZIEL: Finde die Fehlerursache (z.B. Batterie leer) und bestimme die geeignete Reparatur (z.B. Batterie aufladen).

- CBR-Strategie:
 - ◆ Ein Fall beschreibt eine bestimmte Diagnosesituation und enthält
 - eine Beschreibung der aufgetretenen Symptome
 - eine Beschreibung des Fehlverhaltens und seiner Ursache(n)
 - eine Beschreibung der Problemlösung (Fehlerbehebung)
 - ◆ Speichere eine Kollektion aller bekannten Fälle in einer Fallbasis.
 - ◆ Finde die Fälle, die zur aktuellen Problemstellung möglichst ähnlich sind, und verwende deren Lösungsteile.

Was ist ein Fall ?

- Ein Fall beschreibt eine spezifische diagnostische Situation.
- Ein Fall beinhaltet verschiedene Merkmale und deren spezifische Werte in dieser Situation.



Was ist ein Fall?

- Ein Fall beschreibt eine bestimmte Diagnosesituation.
- Fälle sind voneinander unabhängig.

Fall 1

Problem (Symptome)

- **Problem:** Frontlicht funktioniert nicht
- **Fahrzeug:** Vectra C
- **Baujahr:** 2002
- **Batteriespannung:** 13,6 V
- **Zustand Birnen:** OK
- **Zustand Schalter:** OK

Lösung

- **Diagnose:** Sicherung vorne defekt
- **Behebung:** Ersetze Sicherung vorne

Fall 2

Problem (Symptome)

- **Problem:** Frontlicht funktioniert nicht
- **Fahrzeug :** Corsa
- **Baujahr :** 2001
- **Batteriespannung:** 12,8 V
- **Zustand Birnen:** schadhaft
- **Zustand Schalter:** OK

Lösung

- **Diagnose:** Birnen defekt
- **Behebung:** Ersetze Birnen

Wie löst man ein neues Problem (1)?

- Ein neues Problem taucht auf. Wir machen verschiedene Beobachtungen in einer neuen Situation. Diese Beobachtungen definieren ein neues Problem.
- Es müssen nicht alle Merkmale bekannt sein.
- Anmerkung: Das neue Problem ist ein Fall ohne Lösungsteil.

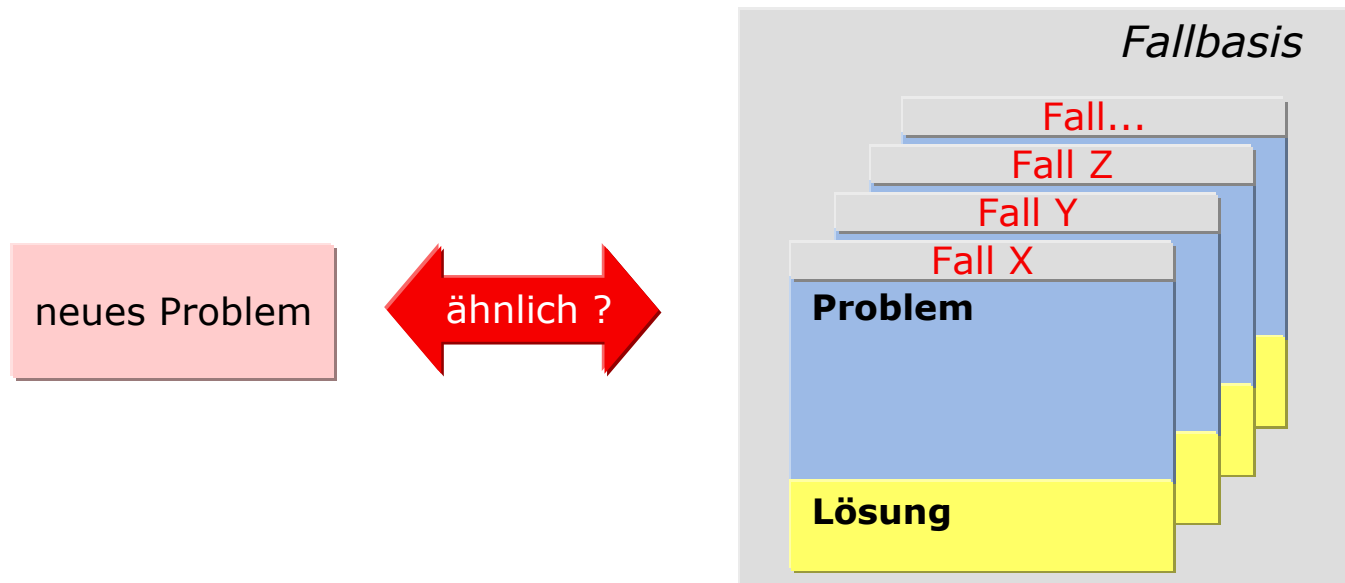
Neuer Fall

Problem (Symptome)

- **Problem:** *Bremslicht funktioniert nicht*
- **Fahrzeug:** *Zafira*
- **Baujahr:** *2000*
- **Batteriespannung:** *12,5 V*
- **Zustand Birnen:** *OK*

Wie löst man ein neues Problem (2)?

- Ähneln der neue Fall einem bestehenden Fall?
- Wie kann man Fälle hinsichtlich ihrer Ähnlichkeit anordnen?
- Anmerkung: Ähnlichkeit ist das zentrale Konzept des CBR !
- Die Ähnlichkeit von Fällen basiert auf der Ähnlichkeit relevanter Merkmalsausprägungen dieser Fälle.
- Die Bedeutung der verschiedenen Merkmale ist entscheidend.



Bestimmung der Ähnlichkeit (1)

- Zuordnung von Ähnlichkeiten für die Ausprägungen der Merkmale.
- Der Grad der Ähnlichkeit wird durch reelle Zahlen zwischen 0 (keine Ähnlichkeit) and 1 (sehr hohe Ähnlichkeit) ausgedrückt.
- Beispiele:

Merkmal: **Problem**

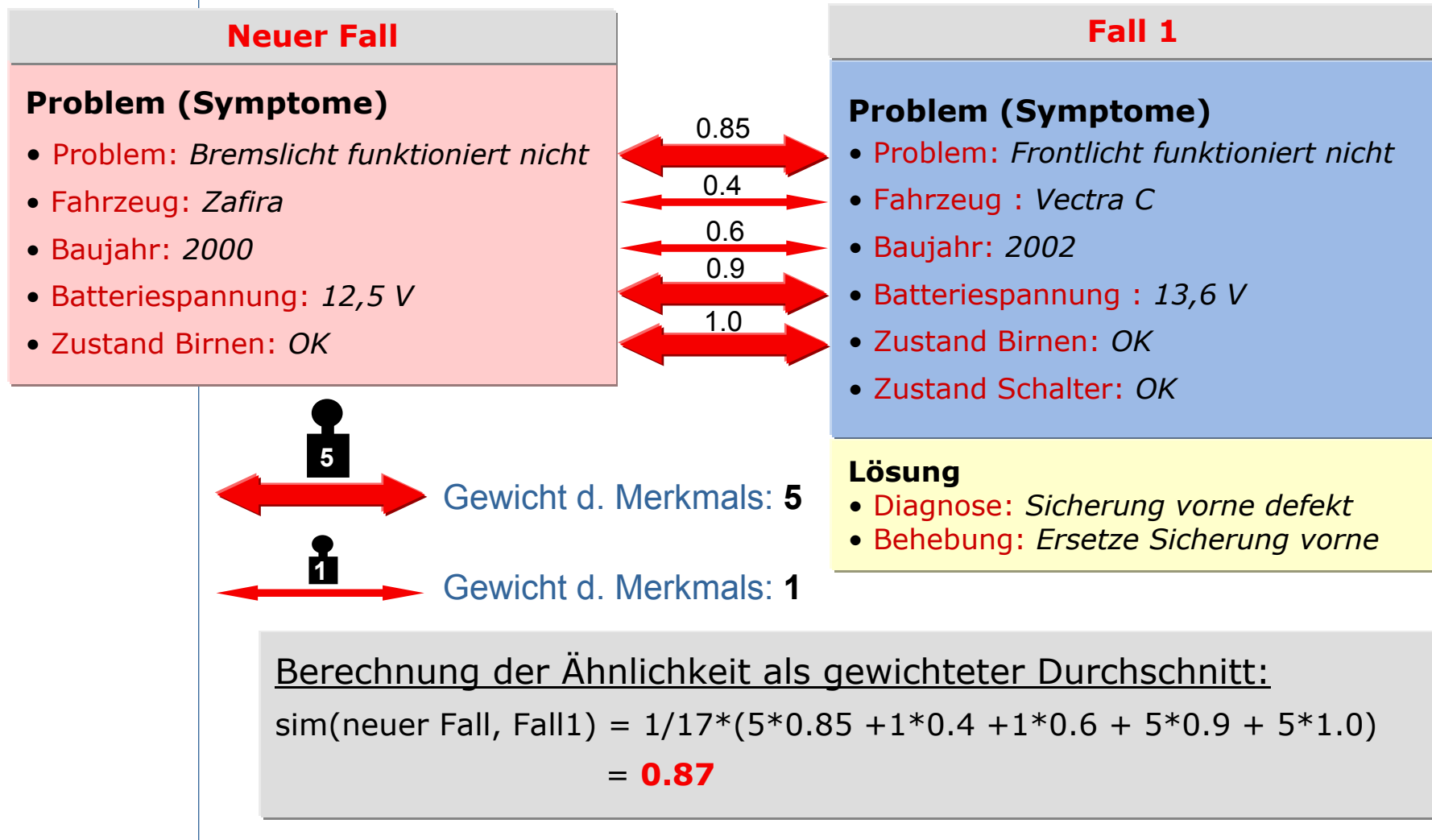
Frontlicht funktioniert nicht \longleftrightarrow $0,85$ \longleftrightarrow Bremslicht funktioniert nicht
Frontlicht funktioniert nicht \longleftrightarrow $0,35$ \longleftrightarrow Motor startet nicht

Merkmal: **Batteriespannung** (abhängig von der Differenz)

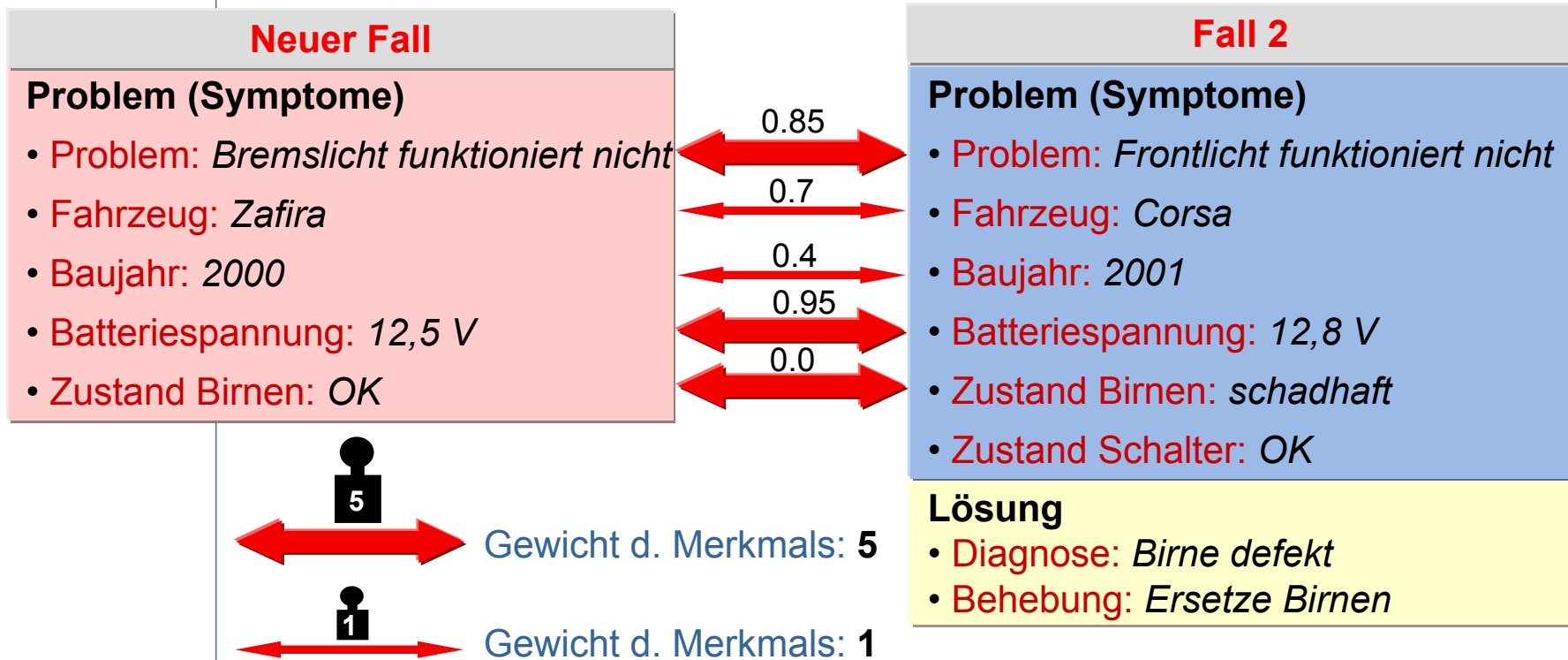
12,8 V \longleftrightarrow $0,9$ \longleftrightarrow 13,6 V
12,8 V \longleftrightarrow $0,1$ \longleftrightarrow 6,5 V

- Verschiedene Merkmale haben unterschiedliche Gewichte.
Hohes Gewicht: **Problem, Batteriespannung, ...**
Niedriges Gewicht: **Fahrzeug, Baujahr, ...**

Ähnlichkeit des neuen Falls mit Fall 1



Ähnlichkeit des neuen Falls mit Fall 2



Berechnung der Ähnlichkeit als gewichteter Durchschnitt:

$$\text{sim}(\text{neuer Fall}, \text{Fall2}) = 1/17 * (5 * 0.85 + 1 * 0.7 + 1 * 0.4 + 5 * 0.95 + 5 * 0) = \mathbf{0.58}$$

Fall 1 ist ähnlicher wg. des Merkmals *Zustand Birnen*

Wiederverwendung der Lösung von Fall 1

- Wie wirken sich Unterschiede in der Problemstellung auf die Lösung aus?
- Wenn die Diagnose korrekt ist, dann speichere den neuen Fall in der Fallbasis.

Neuer Fall

Problem (Symptome)

- **Problem:** Bremslicht funktioniert nicht
- **Fahrzeug:** Zafira
- **Baujahr:** 2000
- **Batteriespannung:** 12,5 V
- **Zustand Birnen:** OK

Fall 1

Problem (Symptome)

- **Problem:** Frontlicht funktioniert nicht
- **Fahrzeug:** Vectra C
- **Baujahr:** 2002
- **Batteriespannung:** 13,6 V
- **Zustand Birnen:** OK
- **Zustand Schalter:** OK

Lösung

- **Diagnose:** Sicherung vorne defekt
- **Behebung:** Ersetze Sicherung vorne



Adap-
tion
der
Lösung



Neue Lösung

- **Diagnose:** Sicherung hinten defekt
- **Behebung:** Ersetze Sicherung hinten

Speicherung des neuen Falles

- Wenn die Diagnose korrekt ist:
- Speichere den neuen Fall.

Fall 3

Problem (Symptome)

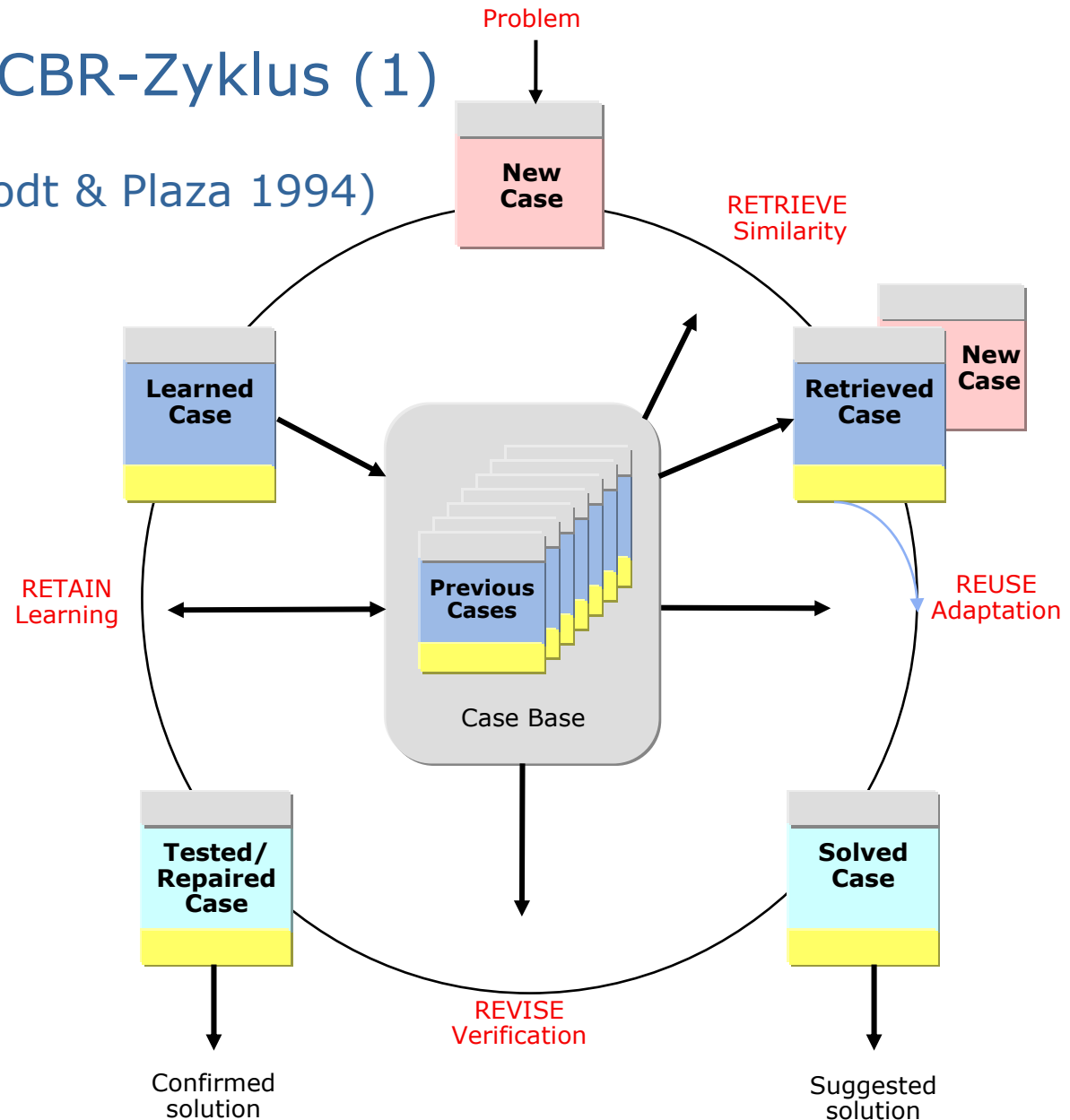
- **Problem:** *Bremslicht funktioniert nicht*
- **Fahrzeug:** *Zafira*
- **Baujahr:** *2000*
- **Batteriespannung:** *12,5 V*
- **Zustand Birnen:** *OK*
- **Zustand Schalter:** *OK*

Lösung

- **Diagnose:** *Sicherung hinten defekt*
- **Behebung:** *Ersetze Sicherung hinten*

Der CBR-Zyklus (1)

(Aamodt & Plaza 1994)



Der CBR-Zyklus (2)

- Retrieval-Phase: Nach Präsentation eines neuen Problems wird zunächst ein geeigneter Fall selektiert.
- Reuse-Phase: Dieser Fall wird wiederverwendet, entweder durch Übernahme oder durch Adaptation der Lösung.
- Revise-Phase: Die neue Lösung wird auf Praktikabilität & Korrektheit geprüft und ggf. revidiert.
- Retain-Phase: Die neue Lösung wird in die Fallbasis eingebracht, im einfachsten Fall durch Speicherung eines neuen Falls.
- Der CBR-Zyklus enthält nur die Basisaktivitäten. Hinzukommen können etwa Aktivitäten wie
 - ◆ Lernen: Die neuen Erfahrungen können dazu dienen, die Adaptationsregeln oder das Ähnlichkeitsmaß zu verbessern.
 - ◆ Wartung der Fallbasis:
 - Entfernen (Vergessen) nicht mehr benötigter Fälle
 - sonstige Wartungsarbeiten

Formalisierung von Fällen

- Gegeben seien zwei Mengen P (Probleme) & L (Lösungen)
- Def.: Ein Fall F ist ein geordnetes Paar $F = (p, l)$ mit $p \in P$ und $l \in L$. Eine Anfrage ist ein Element $p \in P$. Durch Verwendung des Symbols „?“ läßt sich eine Anfrage als Fall $(p, ?)$ darstellen.
- Def.: Eine Fallbasis ist ein geordnetes Paar (FB, S) , wobei FB eine Menge von Fällen $\{F_1, \dots, F_n\}$ und S eine Struktur über FB ist.
- Einige Anmerkungen:
 - ◆ S repräsentiert im wesentlichen eine Indexierung.
 - ◆ Ein Fall könnte auch als Tripel (p, l, k) dargestellt werden, wobei k ein Kommentar sein soll, der Hinweise über Lösungswege und sonstige Kontextinformation enthalten könnte.

Repräsentation von Fällen

- Es gibt keine à priori präferierte Repräsentationsform von Fällen.
- Einfache, flache Merkmalsliste (Attribut-Wert-Paare)
 - ◆ Fälle – Problem- u. Lösungsteil – werden durch Attribut-Wert-Paare dargestellt, z.B. $F=(a_1, \dots, a_n) \in T_1 \times \dots \times T_n$, wobei T_i den Wertebereich des Attributs a_i bezeichnet.
 - ◆ Die Menge der Attribute kann fest oder variabel sein.
 - ◆ Als Ausprägung eines Attributs kann auch „unbekannt“ zugelassen werden.
- Objektorientierte Repräsentation
 - ◆ Ein Fall ist eine Kollektion von Objekten (Instanzen von Klassen) im OO-Sinne.
 - ◆ Insbesondere für Darstellung komplexer und stark strukturierter Sachverhalte erforderlich
- Weitere Darstellungen
 - ◆ Prädikatenlogik, Frames, Graphen, ...

Bsp. für eine objektorient. Fallrepräsentation

- Bsp. aus der med. Diagnose für eine objektorient. Repräsentation mit der Raptor Diagnostic Suite:
 - ◆ Der Fall „Schweinegrippe“ hat hier 7 Feature-Objekte, dem jeweils ein Test-Objekt zugeordnet ist, sowie ein Lösungs-/Behandlungsobjekt.

Fall →
 Feature →
 Test/Prüfung →
 Lösung/
 Behandlung →

The screenshot displays the Raptor Diagnostic Suite interface. On the left, a tree view shows the case structure for 'Grippaler Infekt - Schweinegrippe'. The tree includes nodes for 'Anfangsverdacht', 'Verdacht auf Grippe', 'Körpertemperatur', 'Husten - Niesen', 'Nase Augen', 'Atemtätigkeit', 'Entzündungen', and 'Nahrungsaufnahme'. Each node is associated with a specific test or treatment object. A legend on the left side of the slide maps colors to these categories: black for the case, blue for features, red for tests/examinations, and green for solutions/treatments.

The right pane shows the 'Term' details for 'Körpertemperatur'. It includes fields for 'Name', 'Description', and 'Details'. Below this, the 'Weight' is set to 'High_normal' with a value of 8. The 'Similarity Function' section shows a graph for 'Check Körpertemperatur' with a minimum input value of 36.0 and a maximum input value of 42.0. The graph plots 'Sim' (Similarity) on the y-axis (ranging from -0.4 to 1.0) against an input value on the x-axis. The curve shows a sharp increase in similarity as the input value approaches the maximum, with a specific point marked at 0.9.

Feature-Objekt Körpertemperatur mit lokaler Ähnlichkeitsfunktion, hier am Bsp. der Diagnose-Software „Raptor Diagnostic Suite“.